A class of hierarchical graphs as topologies for interconnection networks*

Pao-Lien Lai[1]†, Hong-Chun Hsu[2],
Chang-Hsiung Tsai[1], and Iain A. Stewart[3]

[1]Department of Computer and Information Engineering,
National Dong Hwa University, Hualien, Taiwan 974, R.O.C.
[2]Department of Medical Informatics, Tzu Chi University,
Hualien, Taiwan 970, R.O.C.
[3]Department of Computer Science, Durham University,
Science Labs, South Road, Durham DH1 3LE, U.K.

Abstract

We study some topological and algorithmic properties of a recently defined hierarchical interconnection network, the hierarchical crossed cube $HCC(k, n)$, which draws upon constructions used within the well-known hypercube and also the crossed cube. In particular, we study: the construction of shortest paths between arbitrary vertices in $HCC(k, n)$; the connectivity of $HCC(k, n)$; and one-to-all broadcasts in parallel machines whose underlying topology is $HCC(k, n)$ (with both one-port and multi-port store-and-forward models of communication). Moreover, some of our proofs are applicable not just to hierarchical crossed cubes but to hierarchical interconnection networks

*This work was supported in part by the National Science Council of the Republic of China under Contract NSC 97-2221-E-259-020.

†Correspondence to: Assistant Professor Pao-Lien Lai, Fax: 886-3-8634053 e-mail: baolein@mail.ndhu.edu.tw.

formed by replacing crossed cubes with other families of interconnection networks. As such, we provide a generic construction with accompanying generic results relating to some topological and algorithmic properties of a wide range of hierarchical interconnection networks.

keywords: hierarchical interconnection networks; hierarchical crossed cubes; hypercubes; crossed cubes; routing; broadcasting; connectivity; diameter; twisted cubes; Möbius cubes.

1 Introduction

The choice of interconnection network is crucial in the design of a distributed-memory multiprocessor. As to which network is chosen depends upon a number of factors relating to the topological, algorithmic, and communication properties of the network and the types of problems to which the resulting computer is to be applied. There is no one optimum interconnection network and a plethora of interconnection networks have been proposed, each with different qualities which vary according to the parameter of interest. For example: K. Chi et al. and S. Zhou recently studied network-coding-based multicast networks in [5] and a class of arc-transitive cayley interconnection networks in [27], respectively.

Hierarchical interconnection networks are, roughly speaking, networks whose edges are partitioned into hierarchies, with each hierarchy defined according to some specific (previously studied) interconnection network. As such, they usually involve a mix of concepts relating to different existing interconnection networks. Hierarchical interconnection networks often have the following structure. The vertices of the network are first partitioned into groups of vertices, with the vertices of each group interconnected according to some prescribed topology. The edges used in this 'layer' of the network are often called internal edges. Next, the vertices of the network (sometimes not all of them) are partitioned in some alternative way and the vertices of each resulting group are interconnected according to some possibly different prescribed topology. The edges used in this layer of the network are often called the external edges. For example: in [8] the two-level binary hypercube-based hierarchical interconnection network is defined where there are 2^D collections of d-dimensional hypercubes with unique vertices in each hypercube forming a set of vertices that are interconnected as a D-dimensional hypercube; in [13] the hierarchical cubic network is defined where 2^n n-dimensional hypercubes are joined so that each vertex in an n-dimensional hypercube is joined to exactly one vertex from some other n-dimensional hypercube; and in [20] the hierarchical hypercube network is defined where 2^{2^m} m-dimensional hypercubes are joined so that each vertex in an m-dimensional hypercube is joined to exactly one vertex from some other m-dimensional hypercube. There are many other existing hierarchical networks including those developed and studied in [2, 7, 9, 11, 12, 14, 19, 21, 22, 23, 25, 26].

As remarked in [25], hierarchical interconnection networks are appealing because: parallel machines with an underlying hierarchical interconnection network topology can be easily expanded so that changes to both the hardware configuration and the communication software of each processor can be minimized; in comparison with some non-hierarchical interconnection networks, such as the hypercube, they can integrate more vertices yet still use the same number of edges; they can integrate the positive features of two or more (non-hierarchical) networks so as to minimize the negative features; and they can support new hybrid computer architectures utilizing both optical and electronic technologies (specifically, processors are partitioned into groups where electronic interconnects are used to connect processors within the same group, while optical interconnects are used for inter-group communication).

A new hierarchical interconnection network, the hierarchical crossed cube $HCC(k, n)$, was proposed in [16]. The hierarchical crossed cube draws upon constructions used within the well-known hypercube[22] and also the crossed cube (a variation of the hypercube as proposed by Efe [11, 12]). In this paper, we study some topological and algorithmic properties of $HCC(k, n)$. In particular, we study: the construction of shortest paths between arbitrary vertices in $HCC(k, n)$; the connectivity of $HCC(k, n)$; and one-to-all broadcasts in parallel machines whose underlying topology is $HCC(k, n)$ (where these machines have one-port or multi-port store-and-forward models of communication). These properties are absolutely fundamental when networks are to be used to inter-connect processors within a distributed-memory multiprocessor. This paper subsumes the results in [16] (we provide improved proofs of these results) and includes new results relating to one-to-all broadcasts. Moreover, some of our proofs are applicable not just to hierarchical crossed cubes but to hierarchical interconnection networks formed by replacing crossed cubes with other families of interconnection networks. As such, we provide a generic construction with accompanying generic results relating to some topological and algorithmic properties of a wide range of hierarchical interconnection networks.

2 Preliminary definitions

In this section we provide definitions relating to hierarchical crossed cubes (first defined and considered in [16]). For definitions of relevant concepts from graph theory and interconnection networks we refer the reader to [24].

As we shall see, the construction of hierarchical crossed cubes is built around those of hypercubes and crossed cubes. The *n-dimensional hypercube* Q_n is possibly the most ubiquitous interconnection network and the related research [1, 10] is still active. Its vertex set is $\{0, 1\}^n$ and there is an edge joining two vertices if, and only if, their names differ in exactly one bit position. Of relevance to us is the fact that the shortest path joining any two vertices of the n-dimensional hypercube is the *Hamming distance* between the two vertices; that is, the number of bit positions where the names of the vertices differ. We denote the length of a shortest path joining any two distinct vertices u and v of any connected graph G by $d_G(u, v)$, and say that the *distance* between u and v is $d_G(u, v)$, with the *diameter* of G being the maximum from

$$\{d \ : \ \text{there exist vertices } u \text{ and } v \text{ in } G \text{ such that the distance between } u \text{ and } v \text{ is } d\}.$$

Consequently, the diameter of Q_n is n. The *connectivity* of a graph G is the minimum number of vertices that have to be removed from G (along with their adjacent edges) so as to produce a disconnected graph. By Menger's Theorem (see [24]), the connectivity of a graph G is equal to the minimum, taken over all pairs of distinct vertices, of the maximum number of vertex-disjoint paths joining the two vertices (where a collection of paths joining vertices x and y is *vertex-disjoint* if no vertex, apart from x and y, appears on more than one path). Moreover, it is trivial to see that if a graph G has connectivity κ, x is a vertex of G, and S is a subset of κ distinct vertices each different from x, then there are κ vertex-disjoint paths joining the vertices in S to x in G. The n-dimensional hypercube is well-known to have connectivity n (see, for example, [24]).

The *n-dimensional crossed cube* CQ_n is a variant of the n-dimensional hypercube. Like the n-dimensional hypercube, its vertex set is $\{0, 1\}^n$. However, the definition of the edges of CQ_n is more involved. We say that

u_2u_1 and v_2v_1, where $u_1, u_2, v_1, v_2 \in \{0,1\}$, are *pair related* if $(u_2u_1, v_2v_1) \in \{(00,00),(10,10),(01,11),(11,01)\}$. The 1-dimensional crossed cube CQ_1 consists of a solitary edge. The n-dimensional crossed cube CQ_n is defined recursively and is built from two disjoint copies of an $(n-1)$-dimensional crossed cube, CQ^0_{n-1} and CQ^1_{n-1}, where the name of any vertex in CQ^i_{n-1} is that of the corresponding vertex from CQ_{n-1} (that is, a bit-string of length $n-1$) prefixed with the bit i, for $i = 0, 1$. There are additional edges joining vertices in CQ^0_{n-1} to vertices in CQ^1_{n-1}. The vertex $0u_{n-1}u_{n-2}\ldots u_2u_1$ of CQ^0_{n-1} is joined to the vertex $1v_{n-1}v_{n-2}\ldots v_2v_1$ of CQ^1_{n-1} if, and only if,

(i) $u_{n-1} = v_{n-1}$, if n is even;

(ii) $u_{2i}u_{2i-1}$ and $v_{2i}v_{2i-1}$ are pair related, for all i such that $1 \leq i < \lceil \frac{n}{2} \rceil$.

A simple induction yields that CQ_n has $n2^{n-1}$ edges (note that by the definition of CQ_n, every vertex in CQ^0_{n-1} has exactly one neighbour in CQ^1_{n-1}, with CQ_1 consisting of a single edge). The diameter of CQ_n is known to be $\lceil \frac{n+1}{2} \rceil$ [11] (there is also a formula for the distance between any two vertices of CQ_n, in terms of their names as bit-strings [3]) and CQ_n has connectivity n [17].

We are now in a position to give the main definition of this section.

Definition 1 Fix $k, n \geq 1$. The *hierarchical crossed cube* $HCC(k,n)$ has vertex set $\{0,1\}^{k+2n}$. Each vertex of $HCC(k,n)$ is written as $(\mathbf{u}, \mathbf{v}, \mathbf{w})$, where $\mathbf{u} \in \{0,1\}^k$ and $\mathbf{v}, \mathbf{w} \in \{0,1\}^n$ (throughout the paper, bold type denotes a bit-string). The set of edges of $HCC(k,n)$ is partitioned into 2 sets, E_{int} and E_{ext}. The set E_{int} is referred to as the set of *internal* edges, whilst the set E_{ext} is referred to as the set of *external* edges. In more detail,

$$E_{int} = \{((\mathbf{u}, \mathbf{v}, \mathbf{w}), (\mathbf{u}, \mathbf{v}, \mathbf{w}')) : (\mathbf{w}, \mathbf{w}') \text{ is an edge of } CQ_n\}$$

and

$$E_{ext} = \{((\mathbf{u}, \mathbf{v}, \mathbf{w}), (\mathbf{u}', \mathbf{w}, \mathbf{v})) : (\mathbf{u}, \mathbf{u}') \text{ is an edge of } Q_k\}.$$

In effect, $HCC(k,n)$ is formed by taking 2^{k+n} disjoint copies of CQ_n, with $CQ_n(\mathbf{u}, \mathbf{v})$ denoting the copy of CQ_n on the set of vertices $\{(\mathbf{u}, \mathbf{v}, \mathbf{w}) : \mathbf{w} \in \{0,1\}^n\}$ (the edges of these copies of CQ_n form the internal edges). The vertices in these copies of CQ_n are then joined by additional edges (the external edges) whereby the vertices are partitioned into 2^{2n} sets of 2^k vertices, with each set of 2^k vertices joined by edges to form a copy of Q_k.

Consequently, edges lie in the 'internal layer' or the 'external layer'. Clearly, $HCC(k,n)$ has 2^{k+2n} vertices, $n2^{k+2n-1}$ internal edges, and $k2^{k+2n-1}$ external edges, making $(n+k)2^{k+2n-1}$ edges in total. By the definition of $HCC(k,n)$, every vertex has n internal neighbours and k external neighbours, and so $HCC(k,n)$ is $(n+k)$-regular. The graph $HCC(2,n)$ can be visualized as in Fig. 1, where the grey ovals are the copies of CQ_n and the black edges are the external edges. Note that the 'twist' in our definition of the external edges (where the positions of **v** and **w** are swapped) is necessary as otherwise the resulting graph would not be connected.

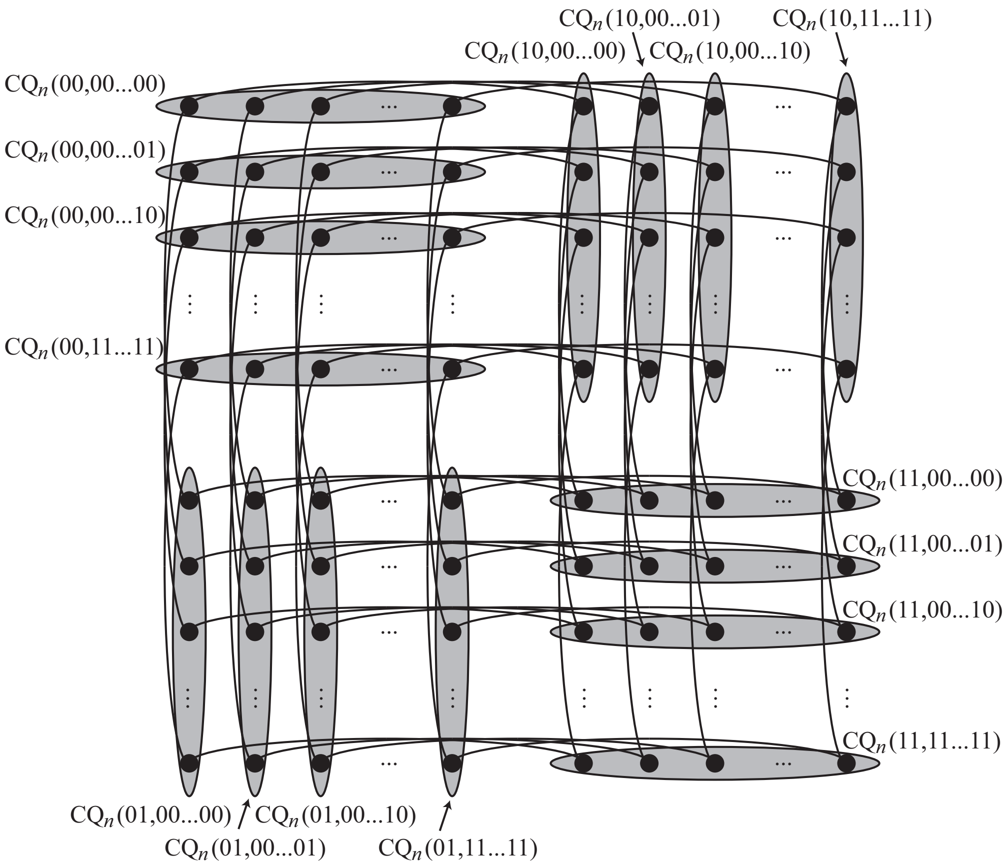

Figure 1. Visualizing $HCC(2,n)$.

We shall write a path of vertices in any graph as $u = u_0 \to u_1 \to u_2 \to \ldots \to u_m = v$, where $u_i \to u_{i+1}$ denotes that an edge joins u_i and u_{i+1}, or as $u \rightarrow^* v$ if we do not need to detail the vertices of the actual path (note that if we write $u \rightarrow^* v$ then it might be the case that $u = v$ and the path is degenerate). However, in $HCC(k,n)$ we write $(\mathbf{u},\mathbf{v},\mathbf{w}) \rightarrow_{CQ_n} (\mathbf{u},\mathbf{v},\mathbf{w}')$ to denote that the edge is an internal edge and $(\mathbf{u},\mathbf{v},\mathbf{w}) \rightarrow_{Q_k} (\mathbf{u}',\mathbf{w},\mathbf{v})$ to denote that the edge is an external edge (we write $\rightarrow^*_{CQ_n}$ and $\rightarrow^*_{Q_k}$ to denote paths of internal or external edges, respectively, of arbitrary lengths where these paths might, in fact, be degenerate). Finally, for any $u \in \{0,1\}$, we denote by $\overline{u}$ the complementary bit to u, and we write $\mathbf{0}$ to denote a tuple of 0's (of some appropriate length).

3 Shortest paths

In this section, we look at determining the shortest path between any two vertices of $HCC(k,n)$, and hence the diameter of $HCC(k,n)$.

Theorem 2 *Let $(\mathbf{u},\mathbf{v},\mathbf{w})$ and $(\mathbf{u}',\mathbf{v}',\mathbf{w}')$ be two distinct vertices of the graph $HCC(k,n)$, where $k,n \geq 1$. Any path ρ joining $(\mathbf{u},\mathbf{v},\mathbf{w})$ and $(\mathbf{u}',\mathbf{v}',\mathbf{w}')$ contains at least $d_{Q_k}(\mathbf{u},\mathbf{u}')$ external edges, unless $\mathbf{u} = \mathbf{u}'$ and $\mathbf{v} \neq \mathbf{v}'$ when it contains at least 2 external edges. Furthermore, the length of any such path ρ is*

- *at least $d_{Q_k}(\mathbf{u},\mathbf{u}') + d_{CQ_n}(\mathbf{v},\mathbf{v}') + d_{CQ_n}(\mathbf{w},\mathbf{w}')$, if $d_{Q_k}(\mathbf{u},\mathbf{u}')$ is even, unless $\mathbf{u} = \mathbf{u}'$ and $\mathbf{v} \neq \mathbf{v}'$ when the length of ρ is at least $2 + d_{CQ_n}(\mathbf{v},\mathbf{v}') + d_{CQ_n}(\mathbf{w},\mathbf{w}')$*
- *at least $d_{Q_k}(\mathbf{u},\mathbf{u}') + d_{CQ_n}(\mathbf{v},\mathbf{w}') + d_{CQ_n}(\mathbf{w},\mathbf{v}')$, if $d_{Q_k}(\mathbf{u},\mathbf{u}')$ is odd.*

Proof Let ρ be any path from $(\mathbf{u},\mathbf{v},\mathbf{w})$ to $(\mathbf{u}',\mathbf{v}',\mathbf{w}')$ in $HCC(k,n)$ where $d_{Q_k}(\mathbf{u},\mathbf{u}')$ is even. Such a path ρ has the form

$$
\begin{aligned}
&(\mathbf{u},\mathbf{v},\mathbf{w}) = (\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_0)\\
&\quad \rightarrow^*_{CQ_n} (\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_1) \rightarrow_{Q_k} (\mathbf{u}_1,\mathbf{w}_1,\mathbf{v}_0) \rightarrow^*_{CQ_n} (\mathbf{u}_1,\mathbf{w}_1,\mathbf{v}_1) \rightarrow_{Q_k} (\mathbf{u}_2,\mathbf{v}_1,\mathbf{w}_1)\\
&\quad \rightarrow^*_{CQ_n} (\mathbf{u}_2,\mathbf{v}_1,\mathbf{w}_2) \rightarrow_{Q_k} (\mathbf{u}_3,\mathbf{w}_2,\mathbf{v}_1) \rightarrow^*_{CQ_n} (\mathbf{u}_3,\mathbf{w}_2,\mathbf{v}_2) \rightarrow_{Q_k} (\mathbf{u}_4,\mathbf{v}_2,\mathbf{w}_2)\\
&\quad \rightarrow^*_{CQ_n} (\mathbf{u}_4,\mathbf{v}_2,\mathbf{w}_3) \rightarrow_{Q_k} (\mathbf{u}_5,\mathbf{w}_3,\mathbf{v}_2) \rightarrow^*_{CQ_n} (\mathbf{u}_5,\mathbf{w}_3,\mathbf{v}_3) \rightarrow_{Q_k} (\mathbf{u}_6,\mathbf{v}_3,\mathbf{w}_3)\\
&\quad \rightarrow^*_{CQ_n} \ldots \rightarrow_{Q_k} (\mathbf{u}_{2m},\mathbf{v}_m,\mathbf{w}_m) \rightarrow^*_{CQ_n} (\mathbf{u}_{2m},\mathbf{v}_m,\mathbf{w}_{m+1}) = (\mathbf{u}',\mathbf{v}',\mathbf{w}'),
\end{aligned}
$$

for some $m \geq 0$ for which $2m \geq d_{Q_k}(\mathbf{u},\mathbf{u}')$. Thus: there is a path

$$\mathbf{w} = \mathbf{w}_0 \rightarrow^* \mathbf{w}_1 \rightarrow^* \mathbf{w}_2 \rightarrow^* \ldots \rightarrow^* \mathbf{w}_m \rightarrow^* \mathbf{w}_{m+1} = \mathbf{w}'$$

in CQ_n; a path

$$\mathbf{v} = \mathbf{v}_0 \to^* \mathbf{v}_1 \to^* \mathbf{v}_2 \to^* \ldots \mathbf{v}_{m-1} \to^* \mathbf{v}_m = \mathbf{v}'$$

in CQ_n; and a path

$$\mathbf{u} = \mathbf{u}_0 \to \mathbf{u}_1 \to \mathbf{u}_2 \to \ldots \to \mathbf{u}_{2m-1} \to \mathbf{u}_{2m} = \mathbf{u}'$$

in Q_k. Consequently, the length of ρ is at least $d_{Q_k}(\mathbf{u}, \mathbf{u}') + d_{CQ_n}(\mathbf{v}, \mathbf{v}') + d_{CQ_n}(\mathbf{w}, \mathbf{w}')$. However, suppose that $\mathbf{u} = \mathbf{u}'$ and $\mathbf{v} \neq \mathbf{v}'$. Any such path ρ must necessarily contain an external edge, and consequently at least two external edges (because $d_{Q_k}(\mathbf{u}, \mathbf{u}')$ is even). Thus, the length of ρ is at least $2 + d_{CQ_n}(\mathbf{v}, \mathbf{v}') + d_{CQ_n}(\mathbf{w}, \mathbf{w}')$.

Let ρ be any path from $(\mathbf{u}, \mathbf{v}, \mathbf{w})$ to $(\mathbf{u}', \mathbf{v}', \mathbf{w}')$ in $HCC(k, n)$ where $d_{Q_k}(\mathbf{u}, \mathbf{u}')$ is odd. Such a path ρ has the form

$$\begin{aligned}
&(\mathbf{u}, \mathbf{v}, \mathbf{w}) = (\mathbf{u}_0, \mathbf{v}_0, \mathbf{w}_0) \\
&\quad \to^*_{CQ_n} (\mathbf{u}_0, \mathbf{v}_0, \mathbf{w}_1) \to_{Q_k} (\mathbf{u}_1, \mathbf{w}_1, \mathbf{v}_0) \to^*_{CQ_n} (\mathbf{u}_1, \mathbf{w}_1, \mathbf{v}_1) \to_{Q_k} (\mathbf{u}_2, \mathbf{v}_1, \mathbf{w}_1) \\
&\quad \to^*_{CQ_n} (\mathbf{u}_2, \mathbf{v}_1, \mathbf{w}_2) \to_{Q_k} (\mathbf{u}_3, \mathbf{w}_2, \mathbf{v}_1) \to^*_{CQ_n} (\mathbf{u}_3, \mathbf{w}_2, \mathbf{v}_2) \to_{Q_k} (\mathbf{u}_4, \mathbf{v}_2, \mathbf{w}_2) \\
&\quad \to^*_{CQ_n} (\mathbf{u}_4, \mathbf{v}_2, \mathbf{w}_3) \to_{Q_k} (\mathbf{u}_5, \mathbf{w}_3, \mathbf{v}_2) \to^*_{CQ_n} (\mathbf{u}_5, \mathbf{w}_3, \mathbf{v}_3) \to_{Q_k} (\mathbf{u}_6, \mathbf{v}_3, \mathbf{w}_3) \\
&\quad \to^*_{CQ_n} \ldots \to_{Q_k} (\mathbf{u}_{2m}, \mathbf{v}_m, \mathbf{w}_m) \to^*_{CQ_n} (\mathbf{u}_{2m}, \mathbf{v}_m, \mathbf{w}_{m+1}) \\
&\quad \to_{Q_k} (\mathbf{u}_{2m+1}, \mathbf{w}_{m+1}, \mathbf{v}_m) \to^*_{CQ_n} (\mathbf{u}_{2m+1}, \mathbf{w}_{m+1}, \mathbf{v}_{m+1}) = (\mathbf{u}', \mathbf{v}', \mathbf{w}'),
\end{aligned}$$

for some $m \geq 0$ for which $2m + 1 \geq d_{Q_k}(\mathbf{u}, \mathbf{u}')$. Thus: there is a path

$$\mathbf{w} = \mathbf{w}_0 \to^* \mathbf{w}_1 \to^* \mathbf{w}_2 \to^* \ldots \to^* \mathbf{w}_m \to^* \mathbf{w}_{m+1} = \mathbf{v}'$$

in CQ_n; a path

$$\mathbf{v} = \mathbf{v}_0 \to^* \mathbf{v}_1 \to^* \mathbf{v}_2 \to^* \ldots \mathbf{v}_m \to^* \mathbf{v}_{m+1} = \mathbf{w}'$$

in CQ_n; and a path

$$\mathbf{u} = \mathbf{u}_0 \to \mathbf{u}_1 \to \mathbf{u}_2 \to \ldots \to \mathbf{u}_{2m} \to \mathbf{u}_{2m+1} = \mathbf{u}'$$

in Q_k. Consequently, the length of ρ is at least $d_{Q_k}(\mathbf{u}, \mathbf{u}') + d_{CQ_n}(\mathbf{v}, \mathbf{w}') + d_{CQ_n}(\mathbf{v}, \mathbf{w}')$. The result follows. $\square$

Corollary 3 *Fix $k, n \geq 1$. Let $(\mathbf{u}, \mathbf{v}, \mathbf{w})$ and $(\mathbf{u}', \mathbf{v}', \mathbf{w}')$ be distinct vertices of $HCC(k, n)$.*

- *Suppose that* $d_{Q_k}(\mathbf{u},\mathbf{u}')$ *is even. If* $\mathbf{u}=\mathbf{u}'$ *and* $\mathbf{v}\neq\mathbf{v}'$ *then we have that* $d_{HCC(k,n)}((\mathbf{u},\mathbf{v},\mathbf{w}),(\mathbf{u}',\mathbf{v}',\mathbf{w}'))$ *is equal to*

  $$2+d_{CQ_n}(\mathbf{v},\mathbf{v}')+d_{CQ_n}(\mathbf{w},\mathbf{w}');$$

  otherwise it is equal to

  $$d_{Q_k}(\mathbf{u},\mathbf{u}')+d_{CQ_n}(\mathbf{v},\mathbf{v}')+d_{CQ_n}(\mathbf{w},\mathbf{w}').$$

- *Suppose that* $d_{q_k}(\mathbf{u},\mathbf{u}')$ *is odd. Then* $d_{HCC(k,n)}((\mathbf{u},\mathbf{v},\mathbf{w}),(\mathbf{u}',\mathbf{v}',\mathbf{w}'))$ *is equal to*

  $$d_{Q_k}(\mathbf{u},\mathbf{u}')+d_{CQ_n}(\mathbf{v},\mathbf{w}')+d_{CQ_n}(\mathbf{w},\mathbf{v}').$$

In consequence, the graph $HCC(k,n)$ *has diameter* $\max\{2,k\}+2\lceil\frac{n+1}{2}\rceil$.

Proof Let $(\mathbf{u},\mathbf{v},\mathbf{w})$ and $(\mathbf{u}',\mathbf{v}',\mathbf{w}')$ be distinct vertices of $HCC(k,n)$. Suppose that:

$$\mathbf{u}=\mathbf{u}_0\rightarrow\mathbf{u}_1\rightarrow\mathbf{u}_2\rightarrow\ldots\rightarrow\mathbf{u}_i=\mathbf{u}'$$

is a shortest path in Q_k from $\mathbf{u}$ to $\mathbf{u}'$;

$$\mathbf{v}=\mathbf{v}_0\rightarrow\mathbf{v}_1\rightarrow\mathbf{v}_2\rightarrow\ldots\rightarrow\mathbf{v}_j=\mathbf{v}'$$

is a shortest path in CQ_n from $\mathbf{v}$ to $\mathbf{v}'$; and

$$\mathbf{w}=\mathbf{w}_0\rightarrow\mathbf{w}_1\rightarrow\mathbf{w}_2\rightarrow\ldots\rightarrow\mathbf{w}_l=\mathbf{w}'$$

is a shortest path in CQ_n from $\mathbf{w}$ to $\mathbf{w}'$ (of course, any of i, j and l might be 0).

Suppose that $d_{Q_k}(\mathbf{u},\mathbf{u}')$ is even and that it is not the case that $\mathbf{u}=\mathbf{u}'$ and $\mathbf{v}\neq\mathbf{v}'$. Define the path ρ as

$$\begin{aligned}(\mathbf{u},\mathbf{v},\mathbf{w})&=(\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_0)\rightarrow_{CQ_n}(\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_1)\rightarrow_{CQ_n}\ldots\rightarrow_{CQ_n}(\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_l)\\&\rightarrow_{Q_k}(\mathbf{u}_1,\mathbf{w}_l,\mathbf{v}_0)\rightarrow_{CQ_n}(\mathbf{u}_1,\mathbf{w}_l,\mathbf{v}_1)\rightarrow_{CQ_n}\ldots\rightarrow_{CQ_n}(\mathbf{u}_1,\mathbf{w}_l,\mathbf{v}_j)\\&\rightarrow_{Q_k}(\mathbf{u}_2,\mathbf{v}_j,\mathbf{w}_l)\rightarrow_{Q_k}(\mathbf{u}_3,\mathbf{w}_l,\mathbf{v}_j)\rightarrow_{Q_k}\ldots\rightarrow_{Q_k}(\mathbf{u}_{i-2},\mathbf{v}_j,\mathbf{w}_l)\\&\rightarrow_{Q_k}(\mathbf{u}_{i-1},\mathbf{w}_l,\mathbf{v}_j)\rightarrow_{Q_k}(\mathbf{u}_i,\mathbf{v}_j,\mathbf{w}_l)=(\mathbf{u}',\mathbf{v}',\mathbf{w}').\end{aligned}$$

Suppose that $\mathbf{u}=\mathbf{u}'$ and $\mathbf{v}\neq\mathbf{v}'$. Define the path ρ as

$$\begin{aligned}(\mathbf{u},\mathbf{v},\mathbf{w})&=(\mathbf{u},\mathbf{v}_0,\mathbf{w}_0)\rightarrow_{CQ_n}(\mathbf{u},\mathbf{v}_0,\mathbf{w}_1)\rightarrow_{CQ_n}\ldots\rightarrow_{CQ_n}(\mathbf{u},\mathbf{v}_0,\mathbf{w}_l)\\&\rightarrow_{Q_k}(\mathbf{u}'',\mathbf{w}_l,\mathbf{v}_0)\rightarrow_{CQ_n}(\mathbf{u}'',\mathbf{w}_l,\mathbf{v}_1)\rightarrow_{CQ_n}\ldots\rightarrow_{CQ_n}(\mathbf{u}'',\mathbf{w}_l,\mathbf{v}_j)\\&\rightarrow_{Q_k}(\mathbf{u},\mathbf{v}_j,\mathbf{w}_l)=(\mathbf{u}',\mathbf{v}',\mathbf{w}'),\end{aligned}$$

where $\mathbf{u}''$ is any neighbour of $\mathbf{u}$ in Q_k.

Suppose that $d_{Q_k}(\mathbf{u},\mathbf{u}')$ is odd. Define the path ρ as

$$\begin{aligned}(\mathbf{u},\mathbf{v},\mathbf{w}) = (\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_0) &\to_{CQ_n} (\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_1) \to_{CQ_n} \ldots \to_{CQ_n} (\mathbf{u}_0,\mathbf{v}_0,\mathbf{w}_l)\\ &\to_{Q_k} (\mathbf{u}_1,\mathbf{w}_l,\mathbf{v}_0) \to_{CQ_n} (\mathbf{u}_1,\mathbf{w}_l,\mathbf{v}_1) \to_{CQ_n} \ldots \to_{CQ_n} (\mathbf{u}_1,\mathbf{w}_l,\mathbf{v}_j)\\ &\to_{Q_k} (\mathbf{u}_2,\mathbf{v}_j,\mathbf{w}_l) \to_{Q_k} (\mathbf{u}_3,\mathbf{w}_l,\mathbf{v}_j) \to_{Q_k} \ldots \to_{Q_k} (\mathbf{u}_{i-2},\mathbf{w}_l,\mathbf{v}_j)\\ &\to_{Q_k} (\mathbf{u}_{i-1},\mathbf{v}_j,\mathbf{w}_l) \to_{Q_k} (\mathbf{u}_i,\mathbf{w}_l,\mathbf{v}_j) = (\mathbf{u}',\mathbf{w}',\mathbf{v}').\end{aligned}$$

Of course, to obtain a path (of the same length) from $(\mathbf{u},\mathbf{v},\mathbf{w})$ to $(\mathbf{u}',\mathbf{v}',\mathbf{w}')$, we simply work with paths in CQ_n from $\mathbf{v}$ to $\mathbf{w}'$ and from $\mathbf{w}$ to $\mathbf{v}'$ instead of paths from $\mathbf{v}$ to $\mathbf{v}'$ and from $\mathbf{w}$ to $\mathbf{w}'$. The result follows by Theorem 2 and the facts that the diameters of Q_k and CQ_n are k and $\lceil\frac{n+1}{2}\rceil$, respectively. $\square$

4 Connectivity

In this section, we consider the connectivity of $HCC(k,n)$. We begin with $HCC(1,n)$, where $n \geq 1$. We can assume that $n \geq 3$ as given the depictions of $HCC(1,1)$ and $HCC(1,2)$ in Figs. 2 and 3, it is trivial to see that $HCC(1,1)$ and $HCC(1,2)$ have connectivity 2 and 3, respectively.

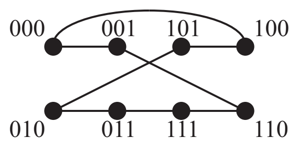

Figure 2. The graph $HCC(1,1)$.

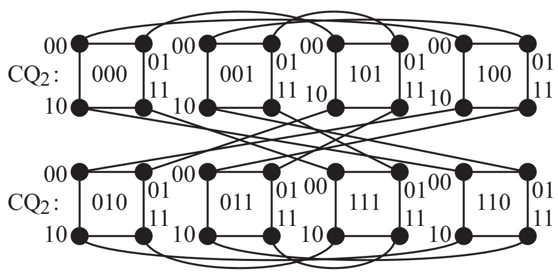

Figure 3. The graph $HCC(1,2)$.

Proposition 4 *Let $n \geq 3$. The graph $HCC(1,n)$ has connectivity $n+1$.*

Proof Let x and y be any two distinct vertices of $HCC(1, n)$. We shall show how $n+1$ vertex-disjoint paths joining x and y can be constructed. There are three essential cases.

Case 1: $x = (0, \mathbf{v}, \mathbf{w})$ and $y = (0, \mathbf{v}, \mathbf{w}')$.

By [12, 17], there are n vertex-disjoint paths in $CQ_n(0, \mathbf{v})$ joining x and y. Also, consider a path $x = (0, \mathbf{v}, \mathbf{w}) \rightarrow_{Q_1} (1, \mathbf{w}, \mathbf{v}) \rightarrow_{CQ_n} (1, \mathbf{w}, \mathbf{v}') \rightarrow_{Q_1} (0, \mathbf{v}', \mathbf{w}) \rightarrow^*_{CQ_n} (0, \mathbf{v}', \mathbf{w}') \rightarrow_{Q_1} (1, \mathbf{w}', \mathbf{v}') \rightarrow_{CQ_n} (1, \mathbf{w}', \mathbf{v}) \rightarrow_{Q_1} (0, \mathbf{v}, \mathbf{w}') = y$, where $\mathbf{v}'$ is a neighbour of $\mathbf{v}$ in CQ_n and where the path $(0, \mathbf{v}', \mathbf{w}) \rightarrow^*_{CQ_n} (0, \mathbf{v}', \mathbf{w}')$ is a path in $CQ_n(0, \mathbf{v}')$ corresponding to some path in CQ_n from $\mathbf{w}$ to $\mathbf{w}'$ (we adopt this denotation of paths throughout this proof). This path from x to y is vertex-disjoint from the other n paths joining x and y.

Case 2: $x = (0, \mathbf{v}, \mathbf{w})$ and $y = (0, \mathbf{v}', \mathbf{w}')$, where $\mathbf{v} \neq \mathbf{v}'$.

Choose n distinct vertices $\{(0, \mathbf{v}, \mathbf{z}_i) : \mathbf{z}_i \notin \{\mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}'\}, i = 1, 2, \ldots, n\}$ in $CQ_n(0, \mathbf{v})$ (note that $n \geq 3$). By [12, 17], there are n vertex-disjoint paths in $CQ_n(0, \mathbf{v})$ joining x with the vertices from $\{(0, \mathbf{v}, \mathbf{z}_i) : \mathbf{z}_i \neq \mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}', i = 1, 2, \ldots, n\}$. Denote the path from x to $(0, \mathbf{v}, \mathbf{z}_i)$ by ρ_i, for $i = 1, 2, \ldots, n$, and consider the path ρ_i extended by the path $(0, \mathbf{v}, \mathbf{z}_i) \rightarrow_{Q_1} (1, \mathbf{z}_i, \mathbf{v}) \rightarrow^*_{CQ_n} (1, \mathbf{z}_i, \mathbf{v}') \rightarrow_{Q_1} (0, \mathbf{v}', \mathbf{z}_i)$. By [12, 17], there exist n vertex-disjoint paths in $CQ_n(0, \mathbf{v}')$ from the vertices of $\{(0, \mathbf{v}', \mathbf{z}_i) : i = 1, 2, \ldots, n\}$ to y. Hence, we clearly have n vertex-disjoint paths in $HCC(1, n)$ from x to y.

Suppose that $\mathbf{w} \neq \mathbf{w}'$. Consider the path: $x = (0, \mathbf{v}, \mathbf{w}) \rightarrow_{Q_1} (1, \mathbf{w}, \mathbf{v}) \rightarrow^*_{CQ_n} (1, \mathbf{w}, \mathbf{v}'') \rightarrow_{Q_1} (0, \mathbf{v}'', \mathbf{w}) \rightarrow^*_{CQ_n} (0, \mathbf{v}'', \mathbf{w}') \rightarrow_{Q_1} (1, \mathbf{w}', \mathbf{v}'') \rightarrow^*_{CQ_n} (1, \mathbf{w}', \mathbf{v}') \rightarrow_{Q_1} (0, \mathbf{v}', \mathbf{w}') = y$, where $\mathbf{v}''$ is a vertex of CQ_n different from $\mathbf{v}$ and $\mathbf{v}'$. Suppose that $\mathbf{w} = \mathbf{w}'$. Consider the path $x = (0, \mathbf{v}, \mathbf{w}) \rightarrow_{Q_1} (1, \mathbf{w}, \mathbf{v}) \rightarrow^*_{CQ_n} (1, \mathbf{w}, \mathbf{v}') \rightarrow_{Q_1} (0, \mathbf{v}', \mathbf{w}) = y$. In both cases, the resulting path from x to y is clearly vertex-disjoint from the other n paths constructed above.

Case 3: $x = (0, \mathbf{v}, \mathbf{w})$ and $y = (1, \mathbf{v}', \mathbf{w}')$.

Sub-case (a): $(\mathbf{v} \neq \mathbf{w}'$ and $\mathbf{w} \neq \mathbf{v}')$ or $(\mathbf{v} = \mathbf{w}'$ and $\mathbf{w} = \mathbf{v}')$.

Choose n distinct vertices $\{(0, \mathbf{v}, \mathbf{z}_i) : \mathbf{z}_i \notin \{\mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}'\}, i = 1, 2, \ldots, n\}$ in $CQ_n(0, \mathbf{v})$ (note that $n \geq 3$). By [12, 17], there are n vertex-disjoint paths joining x with each of the vertices from $\{(0, \mathbf{v}, \mathbf{z}_i) : \mathbf{z}_i \neq \mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}', i = 1, 2, \ldots, n\}$. Denote the path from x to $(0, \mathbf{v}, \mathbf{z}_i)$ by ρ_i, for $i = 1, 2, \ldots, n$. By [12, 17], there are n vertex-disjoint paths joining y with each of the vertices from $\{(1, \mathbf{v}', \mathbf{z}_i) : \mathbf{z}_i \neq \mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}', i = 1, 2, \ldots, n\}$. Denote the path from y to $(1, \mathbf{v}', \mathbf{z}_i)$ by ρ'_i, for $i = 1, 2, \ldots, n$.

For each $i = 1, 2, \dots, n$, extend ρ_i with the path $(0, \mathbf{v}, \mathbf{z}_i) \rightarrow_{Q_1} (1, \mathbf{z}_i, \mathbf{v}) \rightarrow^*_{CQ_n} (1, \mathbf{z}_i, \mathbf{z}_i) \rightarrow_{Q_1} (0, \mathbf{z}_i, \mathbf{z}_i) \rightarrow^*_{CQ_n} (0, \mathbf{z}_i, \mathbf{v}') \rightarrow_{Q_1} (1, \mathbf{v}', \mathbf{z}_i)$ and then with the path ρ'_i. This results in n vertex-disjoint paths.

Suppose that $\mathbf{v} \neq \mathbf{w}'$ and $\mathbf{w} \neq \mathbf{v}'$. The path $x = (0, \mathbf{v}, \mathbf{w}) \rightarrow_{Q_1} (1, \mathbf{w}, \mathbf{v}) \rightarrow^*_{CQ_n} (1, \mathbf{w}, \mathbf{w}') \rightarrow_{Q_1} (0, \mathbf{w}', \mathbf{w}) \rightarrow^*_{CQ_n} (0, \mathbf{w}', \mathbf{v}') \rightarrow_{Q_1} (1, \mathbf{v}', \mathbf{w}') = y$ is vertex-disjoint from the n paths above. The situation can be visualized as in Fig. 4. Suppose that $\mathbf{v} = \mathbf{w}'$ and $\mathbf{w} = \mathbf{v}'$. The path $x = (0, \mathbf{v}, \mathbf{w}) \rightarrow_{Q_1} (1, \mathbf{w}, \mathbf{v}) = y$ is trivially vertex-disjoint from the n paths above.

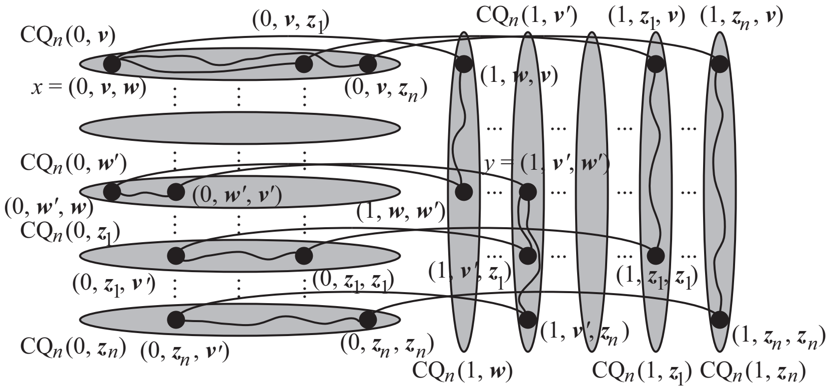

Figure 4. Sub-case $3(a)$ when $\mathbf{v} \neq \mathbf{w}'$ and $\mathbf{w} \neq \mathbf{v}'$.

<u>Sub-case (b)</u>: $\mathbf{v} = \mathbf{w}'$ and $\mathbf{w} \neq \mathbf{v}'$.

Choose $n-1$ distinct vertices $\{(0, \mathbf{v}, \mathbf{z}_i) : \mathbf{z}_i \notin \{\mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}'\}, i = 1, 2, \dots, n-1\}$ in $CQ_n(0, \mathbf{v})$ and set $\mathbf{z}_n = (0, \mathbf{v}, \mathbf{v}')$ (note that $n \geq 3$). By [12, 17], there are n vertex-disjoint paths joining x with each of the vertices from $\{(0, \mathbf{v}, \mathbf{z}_i) : \mathbf{z}_i \neq \mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}', i = 1, 2, \dots, n-1, \text{ and } \mathbf{z}_n = \mathbf{v}'\}$. Denote the path from x to $(0, \mathbf{v}, \mathbf{z}_i)$ by ρ_i, for $i = 1, 2, \dots, n$.

Choose n distinct vertices $\{(1, \mathbf{v}', \mathbf{z}'_i) : \mathbf{z}'_i \neq \mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}', i = 1, 2, \dots, n\}$ in $CQ_n(1, \mathbf{v}')$. By [12, 17], there are n vertex-disjoint paths joining y with each of the vertices from $\{(1, \mathbf{v}', \mathbf{z}'_i) : \mathbf{z}'_i \neq \mathbf{v}, \mathbf{v}', \mathbf{w}, \mathbf{w}', i = 1, 2, \dots, n\}$. Denote the path from y to $(1, \mathbf{v}', \mathbf{z}'_i)$ by ρ'_i, for $i = 1, 2, \dots, n$.

For each $i = 1, 2, \dots, n-1$, extend ρ_i by the path $(0, \mathbf{v}, \mathbf{z}_i) \rightarrow_{Q_1} (1, \mathbf{z}_i, \mathbf{v}) \rightarrow^*_{CQ_n} (1, \mathbf{z}_i, \mathbf{z}'_i) \rightarrow_{Q_1} (0, \mathbf{z}'_i, \mathbf{z}_i) \rightarrow^*_{CQ_n} (0, \mathbf{z}'_i, \mathbf{v}') \rightarrow_{Q_1} (1, \mathbf{v}', \mathbf{z}'_i)$ and then by the path ρ'_i. This results in $n-1$ vertex-disjoint paths.

Consider the path ρ_n extended with the path $(0, \mathbf{v}, \mathbf{v}') \rightarrow_{Q_1} (1, \mathbf{v}', \mathbf{v}) = y$: denote this path by σ. Consider also the path $x = (0, \mathbf{v}, \mathbf{w}) \rightarrow_{Q_1}$

$(1, \mathbf{w}, \mathbf{v}) \rightarrow^*_{CQ_n} (1, \mathbf{w}, \mathbf{z}'_n) \rightarrow_{Q_1} (0, \mathbf{z}'_n, \mathbf{w}) \rightarrow^*_{CQ_n} (0, \mathbf{z}'_n, \mathbf{v}') \rightarrow_{Q_1} (1, \mathbf{v}', \mathbf{z}'_n)$ extended by ρ'_n to obtain a path σ' from x to y. The paths σ and σ' are vertex-disjoint and also vertex-disjoint with all of the $n-1$ paths constructed above. The situation can be visualized as in Fig. 5.

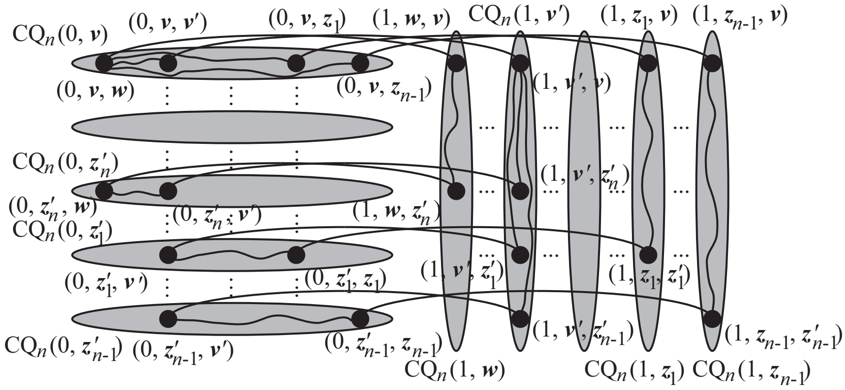

Figure 5. Sub-case $3(b)$ when $\mathbf{v} = \mathbf{w}'$ and $\mathbf{w} \neq \mathbf{v}'$.

Sub-case (c): $\mathbf{v} \neq \mathbf{w}'$ and $\mathbf{w} = \mathbf{v}'$.
Consider the mapping $(0, \mathbf{x}, \mathbf{y}) \mapsto (1, \mathbf{x}, \mathbf{y})$ and $(1, \mathbf{x}, \mathbf{y}) \mapsto (0, \mathbf{x}, \mathbf{y})$ on the vertices of $HCC(1, 1)$. This mapping is clearly an automorphism (see Fig. 2). This reduces this case to Sub-case (b).

The result follows. $\square$

Theorem 5 *For $k, n \geq 1$, $HCC(k, n)$ has connectivity $n + k$.*

Proof Let x and y be distinct vertices in $HCC(k, n)$. We prove by induction on k that there are $n + k$ vertex-disjoint paths from x to y in $HCC(k, n)$. The base case follows by Proposition 4 and the discussion of the cases of $HCC(1, 1)$ and $HCC(1, 2)$, above. Suppose, as our induction hypothesis, that there are $n+k-1$ vertex-disjoint paths joining any two distinct vertices in $HCC(k-1, n)$. For $i = 0, 1$, denote by $H_{k-1}(i)$ the subgraph of $HCC(k, n)$ induced by the vertices of $\{(i\mathbf{u}, \mathbf{v}, \mathbf{w}) : \mathbf{u} \in \{0, 1\}^{k-1}, \mathbf{v}, \mathbf{w} \in \{0, 1\}^n\}$. Clearly, $H_{k-1}(0)$ and $H_{k-1}(1)$ are isomorphic to $HCC(k-1, n)$.
Case 1: $x = (0\mathbf{u}, \mathbf{v}, \mathbf{w}) \in H_{k-1}(0)$ and $y = (1\mathbf{u}', \mathbf{v}', \mathbf{w}') \in H_{k-1}(1)$.
Sub-case (a): x is not adjacent to y in $HCC(k, n)$.

Let $y' = (0\mathbf{u}', \mathbf{w}', \mathbf{v}')$ (and so $y' \neq x$); that is, y' is y's neighbour in $H_{k-1}(0)$. Similarly, define $x' = (1\mathbf{u}, \mathbf{w}, \mathbf{v})$ to be x's neighbour in $H_{k-1}(1)$ (and so $x' \neq y$). By the induction hypothesis applied to $H_{k-1}(0)$, there are $n+k-1$ vertex-disjoint paths $\{\rho_i : i = 1, 2, \ldots, n+k-1\}$ in $H_{k-1}(0)$ joining x and y'. Choose $n+k-2$ of these paths, omitting the path $x \to y'$ if it exists (and so all of the chosen paths have length at least 2). W.l.o.g. let the chosen paths be $\{\rho_i : i = 1, 2, \ldots, n+k-2\}$. Let the penultimate vertex of the path ρ_i be z_i (that is, each z_i is a neighbour of y' and is not equal to x) and let ρ_i' be the path ρ_i truncated at z_i. Furthermore, let the neighbour in $H_{k-1}(1)$ of each z_i be z_i'. By the induction hypothesis applied to $H_{k-1}(1)$, there are $n+k-1$ vertex-disjoint paths joining the vertices of $\{z_i' : i = 1, 2, \ldots, n+k-2\} \cup \{x'\}$ to y. Denote the path from each z_i' to y by σ_i, and the path from x' to y by σ. Hence, by extending each path ρ_i' by the path $z_i \to_{Q_k} z_i'$ and then by the path σ_i, for $i = 1, 2, \ldots, n+k-2$, we obtain $n+k-2$ vertex-disjoint paths from x to y. We also obtain a path from x to y by extending the path $x \to_{Q_k} x'$ by the path σ, which is vertex-disjoint from all of the other paths constructed from x to y. Finally, consider the omitted path from x to y', ρ_{n+k-1}, above, in $H_{k-1}(0)$. We can extend this path by the path $y' \to y$ to obtain yet another path from x to y which is vertex-disjoint from the $n+k-1$ other paths just constructed from x to y. The situation can be visualized as in Fig. 6.

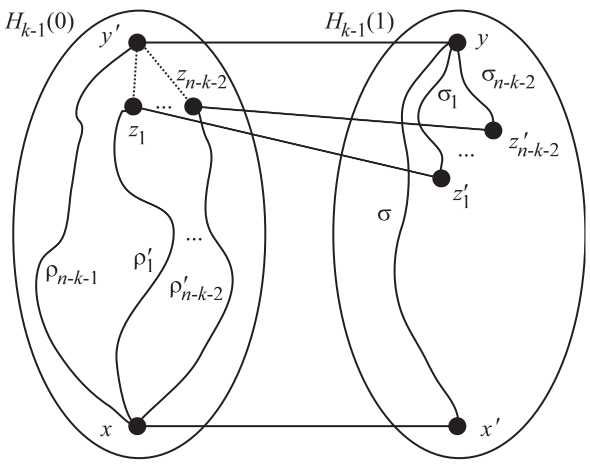

Figure 6. Visualizing the situation in Sub-case 1(a).

Sub-case (b): x is adjacent to y in $HCC(k, n)$.

So, $x = (0\mathbf{u}, \mathbf{v}, \mathbf{w})$ and $y = (1\mathbf{u}, \mathbf{w}, \mathbf{v})$. Let z be any neighbour of x in $H_{k-1}(0)$. By the induction hypothesis applied to $H_{k-1}(0)$, there are $n+k-1$ vertex-disjoint paths from x to z in $H_{k-1}(0)$, one path of which is the path $x \to z$; denote the other paths by $\rho_1, \rho_2, \ldots, \rho_{n+k-2}$. For $i = 1, 2, \ldots, n+k-2$, truncate the path ρ_i at the penultimate vertex z_i and denote it by ρ'_i (so, z_i is a neighbour of z). Let z' be the neighbour of z in $H_{k-1}(1)$, and let z'_i be the neighbour of z_i in $H_{k-1}(1)$, for $i = 1, 2, \ldots, n+k-2$. By the induction hypothesis applied to $H_{k-1}(1)$, there are $n+k-1$ vertex-disjoint paths in $H_{k-1}(1)$ from the vertices of $\{z'_i : i = 1, 2, \ldots, n+k-2\} \cup \{z'\}$ to y: denote the path from z'_i to y by σ_i and denote the path from z' to y by σ. Extend the path ρ'_i by the path $z_i \to_{Q_k} z'_i$ and then by the path σ_i, for each $i = 1, 2, \ldots, n+k-2$. Also, extend the path $x \to z$ by the path $z \to_{Q_k} z'$ and then by the path σ. This yields $n+k-1$ vertex-disjoint paths from x to y in $HCC(k, n)$. Finally, the path $x \to_{Q_k} y$ gives another path. The situation can be visualized as in Fig. 7.

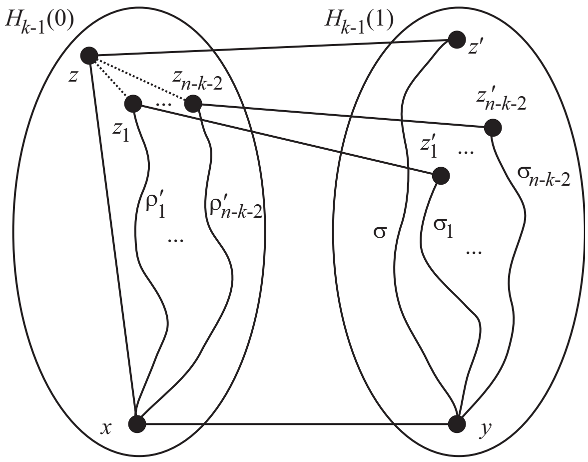

Figure 7. Visualizing the situation in Sub-case $1(b)$.

<u>Case 2</u>: $x = (0\mathbf{u}, \mathbf{v}, \mathbf{w}) \in H_{k-1}(0)$ and $y = (0\mathbf{u}', \mathbf{v}', \mathbf{w}') \in H_{k-1}(0)$ (the case when both x and y are in $H_{k-1}(1)$ is almost identical).

By the induction hypothesis applied to $H_{k-1}(0)$, there are $n+k-1$ vertex-disjoint paths from x to y in $H_{k-1}(0)$. Let x' and y' be the neighbours of x and y in $H_{k-1}(1)$, respectively. There is a path σ from x' to y' in $H_{k-1}(1)$. Hence, the path obtained by extending the path $x \to_{Q_k} x'$ by the path σ and then by the path $y' \to_{Q_k} y$ yields an additional path joining x and y that is vertex-disjoint with the other $n+k-1$ paths. The result in the statement of this theorem now follows by induction as $HCC(k,n)$ is $(n+k)$-regular (and so cannot have connectivity greater than $n+k$). $\square$

Tables 1 and 2 show the major topological characteristics (degrees, connectivities, and diameters) of hypercubes [22], crossed cubes [11, 12], hierarchical hypercubes (HHC) [20], hierarchical cubic networks (HCN)[13], and hierarchical crossed cubes (HCC)[16] for various practical network sizes of 2^l. As seen from the tables, hierarchical crossed cubes compare favourably with these networks, most notably hypercubes and hierarchical cubic networks.

Table 1: The HCC and similar networks compared.

network	vertices	degree	connectivity	diameter
Q_l [22]	2^l	l	l	l
CQ_l [11, 12]	2^l	l	l	$\lceil \frac{l+1}{2} \rceil$
$(m+2^m)$-HHC [20]	2^{m+2^m}	$m+1$	$m+1$	2^{m+1}
$HCN(s,s)$ [13]	2^{2s}	$s+1$	$s+1$	$s+\lfloor \frac{s}{2} \rfloor + 1$
$HCC(k,n)$ [16]	2^{k+2n}	$k+n$	$k+n$	$max\{2,k\}$ $+2\lceil \frac{n+1}{2} \rceil$

Note: $l = m + 2^m = 2s = k + 2n$

5 One-to-all broadcasting

In this section, we examine one-to-all broadcasting in $HCC(k,n)$. Our basic assumption is that we have a synchronous distributed-memory parallel machine M whose underlying topology is that of the graph $HCC(k,n)$; that is, there is a global clock which governs when messages are sent from and received by the processors, which lie at the vertices of $HCC(k,n)$ so that any message is sent along some edge of $HCC(k,n)$. It is always assumed that any sent message is received within the same cycle of the global clock. The machine M is *one-port* if at any time any processor can send at most one message and simultaneously receive at most one message. The machine M is *multi-port* if at any time any processor can send messages to any subset of its neighbours and simultaneously receive messages from any subset of

Table 2: Detailed numerical comparison

desired size 2^l		2^6	2^{11}	2^{20}	2^{37}	2^{70}
Q_l [22]	degree	6	11	20	37	70
	connectivity	6	11	20	37	70
	diameter	6	11	20	37	70
CQ_l [11, 12]	degree	6	11	20	37	70
	connectivity	6	11	20	37	70
	diameter	4	6	11	19	36
$l - HHC$ [20]	m	2	3	4	5	6
	degree	3	4	5	6	7
	connectivity	3	4	5	6	7
	diameter	8	16	32	64	128
$HCN(s,s)$ [13]	s	3		10		35
	degree	4		11		36
	connectivity	4		11		36
	diameter	3		16		53
$HCC(k,n)$ [16]	k	2	3	4	5	6
	n	2	4	8	16	32
	degree	4	6	11	21	38
	connectivity	4	6	11	21	38
	diameter	6	11	14	23	40

Note: $l = m + 2^m = 2s = k + 2n$

its neighbours. A *one-to-all broadcast* in M is a distributed algorithm that, first, constructs a spanning tree within (the underlying topology of) M and, second, disseminates a message from the root of the tree, using the edges of the tree, so that this message is delivered to every other vertex. The aim is usually to complete a one-to-all broadcast in as short a time as possible (where time is measured according to the global clock). We always assume that any message has unit size and that each edge has unit capacity; that is, we have a store-and-forward model of computation.

Intimately related with one-to-all broadcasts is the existence of spanning trees within $HCC(k,n)$, for any spanning tree gives rise to a multi-port algorithm for a one-to-all broadcast in M which takes (global) time equal to the depth of the tree (the message originates at the processor at the root of the tree and is disseminated according to the tree structure). Of course, this requires that the actual tree can be constructed by M, in a distributed fashion, so that any processor has an explicit representation of its parent and children (if any) within the tree. If M is a one-port machine then a spanning tree still gives rise to a one-port algorithm (in fact, numerous such algorithms, depending upon the dissemination strategy) but the resulting algorithm might take time greater than the depth of the tree. Of course, for a universal one-to-all broadcast algorithm we need spanning trees rooted at every vertex of $HCC(k, n)$. We call a spanning tree of a graph a *broadcast tree* if the (rooted) tree is used as the basis of a one-to-all broadcast algorithm.

We shall primarily be concerned with the existence of spanning trees in $HCC(k, n)$ and their structure, in relation to one-to-all broadcasting in a one-port or a multi-port model, rather than the actual (distributed) construction of these trees within some synchronous distributed-memory parallel machine. We shall comment briefly on the actual construction of our trees at the end of the section.

The following theorem shows how broadcast trees in hypercubes and crossed cubes can be composed to form broadcast trees in hierarchical crossed cubes. One problematic aspect of this theorem is that the crossed cube CQ_n is known not to be vertex-symmetric when $n > 4$ [18], although Q_k is vertex-symmetric (see [24]; a graph G is *vertex-symmetric* if given any two distinct vertices u and v of G, there is an automorphism of G mapping u to v). Consequently, our theorem is more involved than it would have been were CQ_n vertex-symmetric.

Theorem 6 *Fix $k \geq 3$ and $n \geq 1$. For each $\mathbf{v} \in \{0, 1\}^n$ and $\mathbf{u} \in \{0, 1\}^{k-2}$, let $T_C^{\mathbf{v}}$ and $T_Q^{\mathbf{u}}$ be broadcast trees in CQ_n and Q_{k-2} rooted at $\mathbf{v}$ and $\mathbf{u}$, respectively. Let: δ_C be the maximum degree of any vertex in any $T_C^{\mathbf{v}}$; δ_Q be the maximum degree of any vertex in any $T_Q^{\mathbf{u}}$; r_C be the maximum degree of the roots in any $T_C^{\mathbf{v}}$; r_Q be the maximum degree of the roots in any $T_Q^{\mathbf{u}}$; β_C be the maximum depth of any tree $T_C^{\mathbf{v}}$; and β_Q be the maximum depth of any tree $T_Q^{\mathbf{u}}$. If $k = 2$ and $n \geq 1$ then define the trees $T_C^{\mathbf{v}}$ and the parameters δ_C, r_C, and β_C as above, and set $\delta_Q = \beta_Q = 0$. For any chosen vertex x of $HCC(k, n)$, there exists a broadcast tree T in $HCC(k, n)$, rooted at x, such that*

- *T has depth at most $\beta_Q + 2\beta_C + 2$*
- *any vertex in T has degree at most $\max\{\delta_Q + r_C + 2, \delta_C + 2\}$.*

Proof We shall begin with the graph $HCC(2, n)$, which can be visualized as in Fig. 1. Fix $u_1, u_2 \in \{0, 1\}$ and $\mathbf{v}, \mathbf{w} \in \{0, 1\}^n$. We shall iteratively build a spanning tree T' in $HCC(2, n)$ rooted at $(u_2u_1, \mathbf{v}, \mathbf{w})$ as follows.

- Initialize the tree T' as the tree $T_C^{\mathbf{w}}$ in $CQ_n(u_2u_1, \mathbf{v})$ rooted at $(u_2u_1, \mathbf{v}, \mathbf{w})$.
- Extend T' by joining each vertex $(u_2u_1, \mathbf{v}, \mathbf{x})$ of T' to its neighbour $(\overline{u}_2u_1, \mathbf{x}, \mathbf{v})$ in $CQ_n(\overline{u}_2u_1, \mathbf{x})$.
- Extend (the new) T' by joining each vertex $(u_2u_1, \mathbf{v}, \mathbf{x})$ of T' to its neighbour $(u_2\overline{u}_1, \mathbf{x}, \mathbf{v})$ in $CQ_n(u_2\overline{u}_1, \mathbf{x})$.
- For each vertex $(\overline{u}_2u_1, \mathbf{x}, \mathbf{v})$ in T', take the tree $T_C^{\mathbf{v}}$ in $CQ_n(\overline{u}_2u_1, \mathbf{x})$ rooted at $(\overline{u}_2u_1, \mathbf{x}, \mathbf{v})$ and extend T' by incorporating this tree $T_C^{\mathbf{v}}$.
- For each vertex $(u_2\overline{u}_1, \mathbf{x}, \mathbf{v})$ in T', take the tree $T_C^{\mathbf{v}}$ in $CQ_n(u_2\overline{u}_1, \mathbf{x})$ rooted at $(u_2\overline{u}_1, \mathbf{x}, \mathbf{v})$ and extend T' by incorporating this tree $T_C^{\mathbf{v}}$.
- For each vertex $(\overline{u}_2u_1, \mathbf{x}, \mathbf{y})$ in T', where $\mathbf{y} \neq \mathbf{v}$, extend T' by joining $(\overline{u}_2u_1, \mathbf{x}, \mathbf{y})$ to its neighbour $(u_2u_1, \mathbf{y}, \mathbf{x})$ of $CQ_n(u_2u_1, \mathbf{y})$.
- For each vertex $(u_2\overline{u}_1, \mathbf{x}, \mathbf{y})$ in T', extend T' by joining $(u_2\overline{u}_1, \mathbf{x}, \mathbf{y})$ to its neighbour $(\overline{u}_2\overline{u}_1, \mathbf{y}, \mathbf{x})$ of $CQ_n(\overline{u}_2\overline{u}_1, \mathbf{y})$.

The resulting tree T' has depth at most $2\beta_C + 2$, maximum degree at most $\delta_C + 2$, and the degree of the root in T' is at most $r_C + 2$. It can be visualized as in Fig. 8, where: for simplicity we have that $u_2u_1 = 00$ and $\mathbf{v} = \mathbf{w} = \mathbf{0}$; the grey ovals are copies of trees $T_C^{\mathbf{x}}$; and the black edges are (external) edges used in T'. Note that the actual tree T' of $HCC(2, n)$ just constructed will, in general, depend upon u_1, u_2, $\mathbf{v}$, and $\mathbf{w}$; so, we refer to it as $T'[u_2u_1, \mathbf{v}, \mathbf{w}]$.

Now let us turn to $HCC(k, n)$, for $k > 2$. For any $\mathbf{x} \in \{0, 1\}^{k-2}$, denote the subgraph of $HCC(k, n)$ induced by the vertices of $\{(\mathbf{x}u_2u_1, \mathbf{v}, \mathbf{w}) : u_1, u_2 \in \{0, 1\}, \mathbf{v}, \mathbf{w} \in \{0, 1\}^n\}$ as $H_2(\mathbf{x})$. Clearly, any such $H_2(\mathbf{x})$ is isomorphic to $HCC(2, n)$.

Fix $\mathbf{u} = u_ku_{k-1} \ldots u_1 \in \{0, 1\}^k$ and set $\mathbf{u}' = u_ku_{k-1} \ldots u_3$. Also, fix $\mathbf{v}, \mathbf{w} \in \{0, 1\}^n$. Let H_Q be the connected component of the subgraph of $HCC(k, n)$ induced by the vertices of $\{(\mathbf{x}u_2u_1, \mathbf{v}, \mathbf{w}), (\mathbf{x}u_2u_1, \mathbf{w}, \mathbf{v}) : \mathbf{x} \in \{0, 1\}^{k-2}\}$ that contains the vertex $(\mathbf{u}, \mathbf{v}, \mathbf{w}) = (\mathbf{u}'u_2u_1, \mathbf{v}, \mathbf{w})$ (note that if $\mathbf{v} \neq \mathbf{w}$ then H_Q has two connected components). Clearly, this connected component of H_Q is isomorphic to Q_{k-2}. Consider the tree $T_Q^{\mathbf{u}'}$ in Q_{k-2} rooted at $\mathbf{u}'$. Initialize the tree T_0 to be the isomorphic copy of $T_Q^{\mathbf{u}'}$ in H_Q rooted at $(\mathbf{u}'u_2u_1, \mathbf{v}, \mathbf{w})$ (note that if $\mathbf{v} \neq \mathbf{w}$ then T_0 is not a spanning tree of H_Q and all edges of T_0 join vertices of the form $(\mathbf{y}u_2u_1, \mathbf{v}, \mathbf{w})$ to vertices of the form $(\mathbf{y}'u_2u_1, \mathbf{w}, \mathbf{v})$).

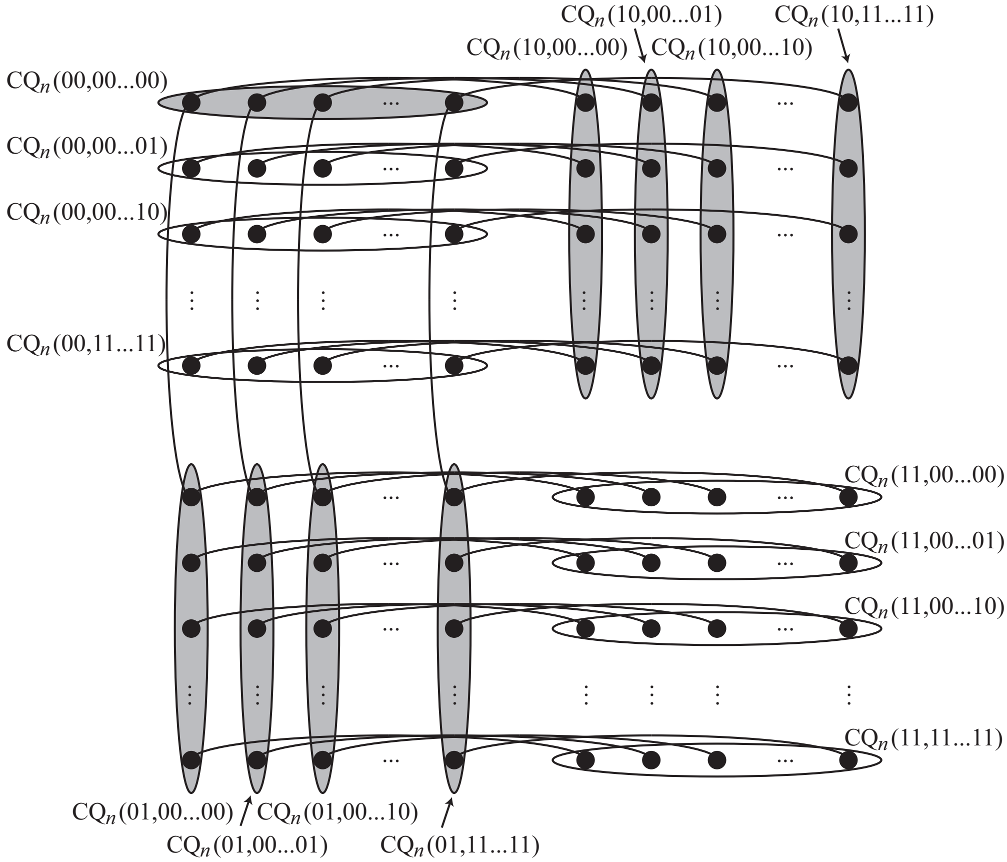

Figure 8. Visualizing T' in $HCC(2, n)$.

Consider some vertex $(\mathbf{x}u_2u_1, \mathbf{v}, \mathbf{w})$ (resp. $(\mathbf{x}u_2u_1, \mathbf{w}, \mathbf{v})$) of T_0. Also, consider the spanning tree $T'[u_2u_1, \mathbf{v}, \mathbf{w}]$ (resp. $T'[u_2u_1, \mathbf{w}, \mathbf{v}]$) in $HCC(2, n)$. From above, $H_2(\mathbf{x})$ is isomorphic to $HCC(2, n)$. So, $H_2(\mathbf{x})$ has an isomorphic copy of $T'[u_2u_1, \mathbf{v}, \mathbf{w}]$ (resp. $T'[u_2u_1, \mathbf{w}, \mathbf{v}]$), denoted $T'[\mathbf{x}u_2u_1, \mathbf{v}, \mathbf{w}]$ (resp. $T'[\mathbf{x}u_2u_1, \mathbf{w}, \mathbf{v}]$), rooted at $(\mathbf{x}u_2u_1, \mathbf{v}, \mathbf{w})$ (resp. $(\mathbf{x}u_2u_1, \mathbf{w}, \mathbf{v})$). Extend T_0 by including all the edges of $T'[\mathbf{x}u_2u_1, \mathbf{v}, \mathbf{w}]$ (resp. $T'[\mathbf{x}u_2u_1, \mathbf{w}, \mathbf{v}]$). Moreover, do this for all vertices of the form $(\mathbf{x}u_2u_1, \mathbf{v}, \mathbf{w})$ or $(\mathbf{x}u_2u_1, \mathbf{w}, \mathbf{v})$ of T_0. Denote the sub-graph so obtained by T

Our new graph T is indeed a tree, for given any $\mathbf{x} \in \{0,1\}^{k-2}$, there is exactly one vertex of T_0 whose first component is $\mathbf{x}u_2u_1$, and $H_2(\mathbf{x})$ and $H_2(\mathbf{x}')$ are vertex-disjoint when $\mathbf{x} \neq \mathbf{x}'$. Moreover, T is a spanning tree of $HCC(k,n)$, and has depth at most $\beta_Q + 2\beta_C + 2$ and degree at most $\max\{\delta_Q + r_C + 2, \delta_C + 2\}$. □

Theorem 6 is particularly flexible in that different broadcast trees, with different properties, can be substituted for the trees $T_C^{\mathbf{v}}$ and $T_Q^{\mathbf{u}}$. Of particular importance are the *binomial trees*. The binomial tree B_0 consists of a solitary vertex which is the root. For $n \geq 1$, the binomial tree B_n is defined recursively by taking two disjoint copies of B_{n-1}, joining their roots by an edge, and making one of these roots the root of B_n. The binomial tree B_n clearly has depth n and 2^n vertices. If we have a binomial tree embedded in (the underlying topology of) a one-port synchronous distributed-memory parallel machine then we can perform a one-to-all broadcast from the root of this tree to all of the processors in the tree in time equal to the depth of the tree (a simple induction shows this, where the first message sent by the root of B_n, say, is to the root of the adjacent sub-tree B_{n-1}). As simple inductions show, both Q_k and CQ_n contain spanning binomial trees which may be rooted at any vertex. To see this, for $k \geq 2$, Q_k is the vertex-disjoint union of two copies of Q_{k-1}, and for $n \geq 2$, CQ_n is the vertex-disjoint union of two copies of CQ_{n-1} (with Q_1 and CQ_1 forming binomial trees, as they both consist of single edges). The induction hypothesis applied to both copies of Q_{k-1} or both copies of CQ_{n-1} yields the result (note that CQ_n contains a binomial tree rooted at any vertex irrespective of the fact that CQ_n is not vertex-symmetric, for $n > 4$).

The following corollary is immediate from Theorem 6 by substituting binomial trees for the trees $T_C^{\mathbf{v}}$ and $T_Q^{\mathbf{u}}$ (along with the fact that, as remarked above, the depth of the binomial tree B_n is n).

Corollary 7 *Fix $k \geq 2$ and $n \geq 1$. For any chosen root, there exists a broadcast tree in the graph $HCC(k,n)$ of depth $k + 2n$.* □

Of course, the broadcast tree in $HCC(k,n)$ in Corollary 7 is not binomial and so it is not immediate that we can use it to perform an efficient one-to-all broadcast in a one-port synchronous distributed-memory parallel machine

M whose underlying topology is $HCC(k,n)$ (note that $k \geq 2$). However, it turns out that we can 'almost achieve' an optimal such algorithm. Let T be the broadcast tree in $HCC(k,n)$ obtained from Corollary 7, rooted at some vertex $(\mathbf{u},\mathbf{v},\mathbf{w})$ of $HCC(k,n)$ and by using binomial trees. Our broadcast algorithm proceeds as follows (we equate $HCC(k,n)$ with the interconnection network of the machine M).

1. Build the tree T within $HCC(k,n)$ with root $(\mathbf{u},\mathbf{v},\mathbf{w})$, where $\mathbf{u} = \mathbf{u}'u_2u_1$.

2. We broadcast our message in $HCC(k,n)$ from the root $(\mathbf{u},\mathbf{v},\mathbf{w})$ and according to the binomial tree of Q_{k-2} so that after $k-2$ time units, every processor in $\{(\mathbf{x}u_2u_1,\mathbf{v}',\mathbf{w}') : \mathbf{x} \in \{0,1\}^{k-2}, (\mathbf{v}',\mathbf{w}') = (\mathbf{v},\mathbf{w}),$ if $d_{Q_{k-2}}(\mathbf{u}',\mathbf{x})$ is even, and $(\mathbf{v}',\mathbf{w}') = (\mathbf{w},\mathbf{v})$, if $d_{Q_{k-2}}(\mathbf{u}',\mathbf{x})$ is odd$\}$ has received the message.

3. As soon as any processor $(\mathbf{x}u_2u_1,\mathbf{v},\mathbf{w})$ or $(\mathbf{x}u_2u_1,\mathbf{w},\mathbf{v})$ has finished sending messages in phase 2, above, it broadcasts the message in $H_2(\mathbf{x})$ (see the proof of Theorem 6 for a definition of $H_2(\mathbf{x})$) as we now explain.

4. As any $H_2(\mathbf{x})$ is isomorphic to $HCC(2,n)$, let us assume that our root processor is the processor $(u_2u_1,\mathbf{v},\mathbf{w})$ of $HCC(2,n)$ (the case when the root is $(u_2u_1,\mathbf{w},\mathbf{v})$ is identical). This root processor begins by broadcasting the message in $CQ_n(u_2u_1,\mathbf{v})$ according to the binomial tree in CQ_n; thus, after at most an additional n time units, every processor in $CQ_n(u_2u_1,\mathbf{v})$ has received the message.

5. As soon as any processor $(u_2u_1,\mathbf{v},\mathbf{y})$ of $CQ_n(u_2u_1,\mathbf{v})$ has finished sending messages in phase 4, above, it sends the message to its external neighbour $(\overline{u}_2u_1,\mathbf{y},\mathbf{v})$ and then to its external neighbour $(u_2\overline{u}_1,\mathbf{y},\mathbf{v})$. These neighbours then embark upon broadcasting the message in $CQ_n(\overline{u}_1u_2,\mathbf{y})$ and $CQ_n(u_2\overline{u}_1,\mathbf{y})$, respectively, according to the binomial tree in CQ_n; thus, after at most an additional $n+2$ time units, every processor in every $CQ_n(\overline{u}_1u_2,\mathbf{y})$ and $CQ_n(u_1\overline{u}_2,\mathbf{y})$ has received the message.

6. Finally, as soon as any processor $(\overline{u}_1u_2,\mathbf{y},\mathbf{x})$ of any $CQ_n(\overline{u}_1u_2,\mathbf{y})$, apart from $(\overline{u}_1u_2,\mathbf{y},\mathbf{v})$, has finished sending messages in phase 5, above, it sends the message to its external neighbour $(u_2u_1,\mathbf{x},\mathbf{y})$. Similarly, as soon as any processor $(u_1\overline{u}_2,\mathbf{y},\mathbf{x})$ of any $CQ_n(u_1\overline{u}_2,\mathbf{y})$ has finished

sending messages in phase 5, above, it sends the message to its external neighbour $(\overline{u}_1\overline{u}_2, \mathbf{x}, \mathbf{y})$. Thus, after an additional 1 time unit, every processor in $HCC(k, n)$ has received the message originating at $(\mathbf{u}, \mathbf{v}, \mathbf{w})$.

The following corollary is immediate.

Corollary 8 *Let M be a one-port synchronous distributed-memory parallel machine whose underlying topology is $HCC(k, n)$, where $k \geq 2$ and $n \geq 1$. For any chosen vertex x, there is a distributed algorithm that performs a one-to-all broadcast from x in M in time $k + 2n + 1$.* $\square$

Note that our one-port one-to-all broadcast in Corollary 8 is 'almost optimal', for consider any one-to-all broadcast in our machine M. A simple induction shows that at any time t, at most 2^t processors have received the message. Thus, any one-to-all broadcast in M necessarily takes time at least $k + 2n$ (when $k \geq 2$).

When $k = 1$ and $n \geq 1$, we can employ almost the same construction in $HCC(k, n)$ as we did in the proof of Theorem 6 (the reader should refer to the 'top half' of Fig. 8) to obtain the following result.

Corollary 9 *Fix $n \geq 1$. For each $\mathbf{v} \in \{0, 1\}^n$, let $T_C^{\mathbf{v}}$ be a broadcast tree in CQ_n rooted at $\mathbf{v}$. Let δ_C be the maximal degree of any vertex in any $T_C^{\mathbf{v}}$ and β_C be the maximal depth of any tree $T_C^{\mathbf{v}}$. For any chosen vertex x of $HCC(1, n)$, there exists a broadcast tree T in $HCC(1, n)$, rooted at x, such that*

- *T has depth at most $2\beta_C + 2$*
- *any vertex in T has degree at most $\delta_C + 1$.*

Choosing our broadcast trees in Corollary 9 to be binomial trees and proceeding similarly to as we did prior to Corollary 8, we immediately obtain the following result.

Corollary 10 *Let M be a one-port synchronous distributed-memory parallel machine whose underlying topology is $HCC(1, n)$, where $n \geq 1$. For any chosen vertex x, there exists a broadcast tree in $HCC(1, n)$, rooted at x and of depth $2n+2$, and a distributed algorithm that performs a one-to-all broadcast in M, according to this tree, in time $2n + 2$.* $\square$

Again, our broadcast algorithm in Corollary 10 is 'nearly optimal' as any one-to-all broadcast in our machine M necessarily takes time at least $2n+1$. We should remark that it might be the case that our one-to-all broadcasts in Corollaries 8 and 10 are, in fact, optimal for $HCC(k,n)$, where $k \geq 1$ and $n \geq 1$, for if could well be the case that such broadcasts in $HCC(k,n)$ can not be undertaken in time $k+2n$, where $k \geq 2$, and $2n+1$, where $k=1$, respectively, irrespective of our lower bound arguments. These questions remain open.

Consider when our machine M is an all-port synchronous distributed-memory parallel machine whose underlying topology is $HCC(k,n)$, where $k \geq 2$. In [12] it is shown that given any vertex x, there is a broadcast tree S_C^x in CQ_n rooted at x and of depth the diameter of CQ_n; that is, S_C^x has depth $\lceil \frac{n+1}{2} \rceil$. Of course, the binomial tree B_k in Q_k has depth k. Consequently, Theorem 6 and Corollary 3 immediately yield the following result.

Corollary 11 *Let M be a multi-port synchronous distributed-memory parallel machine whose underlying topology is $HCC(k,n)$, where $k \geq 2$ and $n \geq 1$. Given any vertex x, there is a distributed algorithm that performs a one-to-all broadcast in M in time $k + 2\lceil \frac{n+1}{2} \rceil$. This algorithm is time-optimal.* □

Similarly, Corollary 9 and Corollary 3 yield the following result.

Corollary 12 *Let M be a multi-port synchronous distributed-memory parallel machine whose underlying topology is $HCC(1,n)$, where $n \geq 1$. Given any vertex x, there is a distributed algorithm that performs a one-to-all broadcast in M in time $2 + 2\lceil \frac{n+1}{2} \rceil$. This algorithm is time-optimal.* □

We end this section with a brief remark concerning the algorithmic construction of the trees used in our one-to-all broadcasts, above, under the assumption that at some point in time a particular processor x in our machine M wishes to undertake a one-to-all broadcast of some particular message. Hitherto, we have not considered the time actually taken to construct these trees (we have simply assumed that these trees are available). Consider broadcasting via a binomial tree in Q_k or in CQ_n. For simplicity, suppose that we wish to broadcast using a binomial tree B_k of Q_k where $x = 00\ldots00$ is to sit at the root of the tree. The processor x would compute its neighbour in dimension 1, namely $00\ldots01$, and send the message to this neighbour. In the next round, both $00\dots00$ and $00\dots01$ would compute their neighbours in dimension 2, namely $00\dots010$ and $00\dots011$, respectively, and send the message to these neighbours. This would continue with the 4 active processors and their neighbours in dimension 3; and so on. Note that the one-to-all broadcast is such that in each round the amount of time spent on deciding which of a processor's neighbours is to be sent the message is constant. Thus, although the eventual binomial tree has a vertex of degree k, no matter what the value of k the one-to-all broadcast can be completed in k rounds and $O(k)$ inclusive time (where 'inclusive time' is to include the time spent in the construction of the tree). An analogous statement can be made as regards CQ_n. Hence, we may assume that the times in Corollaries 8 and 10 refer to inclusive time (subject to replacing the actual times $k+2n+1$ and $2n+2$ with some constant times these numbers). As regards multi-port synchronous distributed-memory parallel machines, we can use Efe's distributed algorithm to embed the tree S_C^y in CQ_n, where the root is y, and all local computation undertaken in any round in order to construct the tree S_C^y takes constant time. Note that in a multi-port model of computation, we may assume that when broadcasting according to a binomial tree only one message is ever sent from any processor in any clock cycle. Hence, again all local computation undertaken in any round in order to construct a binomial tree in Q_k takes constant time. Thus, we may assume that the times in Corollaries 11 and 12 can be taken to mean inclusive time (subject to the same proviso as above). We can make analogous remarks as regards devising shortest-path routing algorithms in our machines (given the shortest-path routing algorithm in [12] and the standard shortest-path routing algorithm in hypercubes).

6 Conclusions

In this paper we have established some basic topological and algorithmic results concerning hierarchical crossed cubes which are hierarchical interconnection networks obtained by fusing hypercubes and crossed cubes. However, we now make a crucial observation: *nowhere throughout this paper have we used any structural properties of crossed cubes apart from the facts that they have diameter $\lceil \frac{n+1}{2} \rceil$, have connectivity n, contain binomial broadcast trees, and contain the broadcast trees as constructed by Efe* [12]. Consequently, we can allow *any* interconnection network to play the role of the crossed cube so long as we substitute the appropriate parameters relating to diameter, connectivity, and so on in any consequent results. We have chosen to present our research via the crossed cube so as to make it concrete and apparent as to the advantages of our general approach.

For example, one could substitute one of the many variants of hypercubes for crossed cubes in our construction such as the twisted cube or the 1-Möbius cubes. It is known that the n-dimensional twisted cube [15] and the n-dimensional 1-Möbius cube [6] have diameter $\lceil \frac{n+1}{2} \rceil$ and connectivity n and $n-1$, respectively [4, 6]. Thus, we would obtain that hierarchical twisted cubes and hierarchical 1-Möbius cubes have diameter $\max\{2, k\} + 2\lceil \frac{n+1}{2} \rceil$, with the former having connectivity $n+k$ and the latter connectivity $n+k-1$. We need not restrict ourselves to substituting only hypercube variants. We can choose any family of interconnection networks to obtain a new hierarchical family to which our results apply. Of course, given appropriate broadcast trees for any new (substitute) interconnection network, we obtain one-to-all broadcast results in the corresponding hierarchical interconnection network.

We end with some proposals as regards further research. Of course, there are many topological and algorithmic properties of hierarchical crossed cubes still to examine, in both fault-free and faulty environments. However, we feel that our generic construction is interesting as it is widely applicable with other interconnection networks replacing crossed cubes. Indeed, we could choose to replace the hypercubes with different interconnection networks too; however, there are no immediate results derivable from those in this paper for such networks as we have explicitly used the internal structure of the hypercube in our proofs. We feel that further investigation of our construction, with other networks replacing hypercubes and crossed cubes, would be beneficial as we can use the 'modular' aspects of the construction to piece together the properties of the component networks in order to establish results for the hierarchical interconnection network. We feel that this line of research is exciting and will yield significant results. As yet, and as far as we are aware, there has only been one attempt, in [7], to provide a systematic consideration of hierarchical interconnection networks, and we feel that such a systematic consideration should be further developed.

References

[1] J. C. Bermond, A. Ferreira, S. Perennes, and J. G. Peters, Neighborhood broadcasting in hypercubes, *SIAM J. Discrete Math.* 4 (2007) 823–843.

[2] S. Campbell, M. Kumar and S. Olariu, The hierarchical cliques interconnection network, *J. Parallel Distrib. Comput.*, 1 (2004) 16–28

[3] C.-P. Chang, T.-Y. Sung and L.-H. Hsu, Edge congestion and topological properties of crossed cubes, *IEEE Trans. Parallel Distrib. Syst.*, 11 (2000) 64–80.

[4] C.-P. Chang, J.-N. Wang and J.-H. Hsu, Topological properties of twisted cube, *Inform. Sciences*, 1-2 (1999) 147–167.

[5] K. Chi, X. Jiang, S. Horiguchi, and M. Guo, Topology design of network-coding-based multicast networks, *IEEE Trans. Parallel Distrib. Syst.*, 5 (2008) 627–640.

[6] P. Cull and S.M. Larson, The Möbius cubes, *IEEE Trans. Comput.*, 5 (1995) 647–659.

[7] P. Cull and S.M. Larson, Smaller diameters in hypercube-variant networks, *Telecommun. Syst.*, 10 (1998) 175–184.

[8] S.P. Dandamudi and D.L. Eager, Hierarchical interconnection networks for multicomputer systems, *IEEE Trans. Comput.*, C-39 (1990) 786–797.

[9] D.-R. Duh, G.-H. Chen and J.-F. Fang, Algorithms and properties of a new two-level network with folded hypercubes as basic modules, *IEEE Trans. Parallel Distrib. Syst.*, 6 (1995) 714–723.

[10] T. Dvorak, and P. Gregor, Partitions of faulty hypercubes into paths with prescribed endvertices, *SIAM J. Discrete Math.* 4 (2008) 1448–1461.

[11] K. Efe, A variation on the hypercube with lower diameter, *IEEE Trans. Comput.*, 11 (1991) 1312–1316.

[12] K. Efe, The crossed cube architecture for parallel computing, *IEEE Trans. Parallel Distrib. Syst.*, 3 (1992) 513–524.

[13] K. Ghose and K.R. Desai, Hierarchical Cubic Networks, *IEEE Trans. Parallel Distrib. Syst.*, 4 (1995) 427–435.

[14] M.M. Hafizur Rahman and S. Horiguchi, HTN: a new hierarchical interconnection network for massively parallel computers, *IEICE Trans. Inf. Syst.*, E86-D (2003) 1479–1486.

[15] P.A.J. Hilbers, R.J.M. Koopman and J.L.A. Van de Snepscheut, The twisted cube, *Proc. Parallel architecture and Languages Europe* (J.W. de Bakker, P.C. Trelearen, eds.), Springer-Verlag, Eindhoven (1987) 152–159.

[16] H.-C. Hsu, C.-H. Tsai, C.-H. Tsai, G.-J. Wu, S.-H. Liu and P.-L. Lai, Hierarchical crossed cube: a new interconnection topology, *Proc. of 25th Workshop on Combinatorial Mathematics and Computation Theory*, Taiwan (2008) 70–79.

[17] P. Kulasinghe, Connectivity of the crossed cube, *Inform. Process. Lett.*, 4 (1997) 221–226.

[18] P. Kulasinghe and S. Bettayeb, Multiply-twisted hypercube with five or more dimensions is not vertex-transitive, *Inform. Process. Lett.*, 1 (1995) 33–36.

[19] J.M. Kumar and L.M. Patnaik, Extended hypercube: a hierarchical interconnection network of hypercube, *IEEE Trans. Parallel Distrib. Syst.*, 3 (1992) 45–57.

[20] Q. M. Malluhi and M. A. Bayoumi, The hierarchical hypercube: a new interconnection topology for massively parallel systems, *IEEE Trans. Parallel Distrib. Syst.*, 1 (1994) 17–30.

[21] M.M.H. Rahman, Y. Inoguchi and S. Horiguchi, Modified hierarchical 3D-torus network, *IEICE Trans. Inf. Syst.*, E88-D (2005) 177–186.

[22] Y. Saad and M. H. Schultz, Topological Properties of Hypercube, *IEEE Tran. Computers*, 7 (1988) 867–872.

[23] W. Shi and P.K. Srimani, Hierarchical star: a new two level interconnection network, *J. Syst. Arch.*, 1 (2005) 1–14.

[24] J. Xu, *Topological Structure and Analysis of Interconnection Networks*, Kluwer, Dordrecht, 2001.

[25] X. Yang, G.M. Megson and D.J. Evans, An oblivious shortest-path routing algorithm for fully connected cubic networks, *J. Parallel Distrib. Comput.*, 10 (2006) 1294–1303.

[26] S.-K. Yun and K.H. Park, Hierarchical hypercube networks (HHN) for massively parallel computers, *J. Parallel Distrib. Comput.*, 2 (1996) 194–199.

[27] S. Zhou, A class of arc-transitive cayley graphs as models for interconnection networks, *SIAM J. Discrete Math.* 2 (2009) 694–714.